A Theory of Recursive Kernel RX Anomaly Detection Algorithm for Hyperspectral Imagery

Chunhui Zhao and Weiwei Deng
Harbin Engineering University, Harbin, Heilongjiang 150001, China

Abstract— With the development of imaging spectroscopy and the improvement of spectral resolution, the ever-expending hyperspectral datasets lead to huge pressure of data storage, downlink transmission and further processing. Real-time processing which requires immediate decision making is greatly desired in hyperspectral anomaly detection. Kernel Reed-Xiaoli detector (KRXD) is a kernel-based nonlinear version of RXD, it achieves better detection accuracy but inferior detection efficiency. This paper developments a new modified KRXD based on progressive line processing that can implement real-time detection in a line-by-line fashion. A new local causal framework is defined to remain the causality of detection system. Aiming at the defect that the complexities of KRXD is high in calculating the detection process, taking advantage of the Woodbury matrix identity and the matrix inversion lemma to recursively update the kernel Gram matrix and its inversion to meet the requirement of rapid processing. Experimental results show the proposed method significantly solves real-time processing problem and keep detection accuracy unchanged compared to the initial algorithm.

1. INTRODUCTION

Hyperspectral imagery (HSI) is constructed as a 3D image cube which contains both spatial information and spectral information. Based on the abundant broadband wavelength and high spectral resolution, it can distinguish subtle differences between ground objects, thereby effectively completing the detection and identification of targets. Hyperspectral detection technology, as one of the hottest topics in current remote sensing information processing and image processing, has been widely used in military, agriculture, forestry and urban planning [1]. In practical application, it is hard to accurately acquired priori information of hyperspectral datasets, therefore, anomaly detection (AD) is of particular importance since it can uncover targets without any prior knowledge.

Reed-Xiaoli detector (RXD) which derived from Generalized Likelihood Ratio Test (GLRT) is regarded as a benchmark for multi/hyperspectral anomaly detection [2]. On the premise that the background datasets are assumed to be Gaussian multivariate distribution, RXD accomplishes the anomaly targets by using the mahalanobis. However, RXD merely utilizes the low-order statistics of hyperspectral data, thus usually leading to the poor detection accuracy. Later, Kwon and Nasrabadi proposed a kernel-based nonlinear version of RXD called kernel RXD (KRXD) [3]. Taking full advantage of nonlinear information between spectral bands, it achieves higher detection probability at the expense of computing time.

In applications of anomaly detection, time delay is usually intolerable, sometimes targets need to be detected effectively to avoid these targets disappearing or being compromised by background clusters. Hence, it becomes a necessary trend to realize the anomaly detection in real-time [4]. Interestingly, the issue of performing KRXD in band interleaved by lines (BIL) has not received much attention in the past. This paper proposes a new real-time KRXD (RT-KRXD) that can detect anomaly targets line by line via BIL.

2. KRXD ALGORITHM

The KRXD which adequately exploits the high-order correlation between hyperspectral bands and obtain a better detection performance is widely used in hyperspectral imagery. Assume that each input signal consisting of J spectral bands be denoted by $\mathbf{x}_i = [x_{i1}, x_{i2}, \ldots, x_{iJ}]^T$. Two Gaussian hypothesis test model in high-dimensional feature space with the nonlinear function Φ for AD that KRXD must distinguish can be expressed as follows:

$$\begin{cases} H_{0\Phi} : \Phi\left(\mathbf{x}\right) = \mathbf{n}_\Phi & \text{(Target absent)} \\ H_{1\Phi} : \Phi\left(\mathbf{x}\right) = \mathbf{a}_\Phi \Phi\left(\mathbf{s}\right) + \mathbf{n}_\Phi & \text{(Target present)} \end{cases} \tag{1}$$

where $\mathbf{a}_\Phi = \mathbf{0}$ under $H_{0\Phi}$ and $\mathbf{a}_\Phi > \mathbf{0}$ under $H_{1\Phi}$, n_Φ and $\Phi(s)$ represent the background clutter and spectral signature of the target in the feature space, respectively. Then the corresponding KRXD

Authorized licensed use limited to: University of Wyoming Libraries. Downloaded on October 11,2021 at 23:51:05 UTC from IEEE Xplore. Restrictions apply.

algorithm is specified by

$$\delta^{KRXD}(\Phi(r)) = (\Phi(r) - \hat{\mu}_{b\Phi})^T \hat{K}_{b\Phi}^{-1} (\Phi(r) - \hat{\mu}_{b\Phi}) \tag{2}$$

where $\hat{\mu}_{b\Phi}$ and $\hat{\mathbf{K}}_{b\Phi}$ are the background data mean vector and the background covariance matrix in the feature space, respectively. Through kernel trick and derivation,

$$\mathbf{k}_{\mathbf{r}}^T = k(\mathbf{r}, \mathbf{X}_w) - \left(\frac{1}{N}\sum_{i=1}^{N} k(\mathbf{r}, \mathbf{x}_i)\right) \mathbf{1}_{1\times N} \tag{3}$$

$$\mathbf{k}_{\hat{\mu}}^T = \left(\frac{1}{N}\sum_{i=1}^{N} k(\mathbf{x}_i, \mathbf{X}_w)\right) - \left(\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} k(\mathbf{x}_i, \mathbf{x}_j)\right) \mathbf{1}_{1\times N} \tag{4}$$

where $\mathbf{X}_w$ is the original background clutter data denoted by $X_w = [x_1, x_2, \dots, x_N]^T$. Finally, the KRXD algorithm can be specified as

$$\delta^{KRXD}(\Phi(\mathbf{r})) = \left(\mathbf{k}_{\mathbf{r}}^{\mathbf{T}} - \mathbf{k}_{\hat{\mu}}^{\mathbf{T}}\right)^{\mathbf{T}} \hat{\mathbf{K}}_w^{-1} \left(\mathbf{k}_{\mathbf{r}}^{\mathbf{T}} - \mathbf{k}_{\hat{\mu}}^{\mathbf{T}}\right) \tag{5}$$

According to the Equation (5), it is clear that the dot product of high dimensional feature space is converted to kernel function of the low-dimensional input space by employing an effective kernel trick. A commonly used kernel is the Gaussian radial basis function (RBF) kernel, which has a translation invariance and better capability of local information retrieval is adopted as the base kernel mapping function:

$$k(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{y}\|^2}{c}\right) \tag{6}$$

where c represents the width of RBF, it is a positive constant.

3. REAL-TIME PROCESSING OF KRXD ALGORITHM

3.1. Sliding Causal Matrix Window

KRXD utilizes the traditional dual window to obtain local background statistical information, the process of it must include future data after the current pixel. Therefore, it cannot guarantee causality, the one requirement of real-time processing. To achieve this goal, this paper develops a new type of sliding causal matrix window as shown in Figure 1. l_a and l_b are defined as width and length of sliding causal matrix window, respectively. Let L_n denotes the nth data line, $w(L_n)$ denotes the current sliding causal matrix window at L_n. When the sliding causal matrix window slides from L_{n-1} to L_n, all the left column pixels in $w(L_{n-1})$ are removed and all the far right column pixels in $w(L_n)$ are filled.

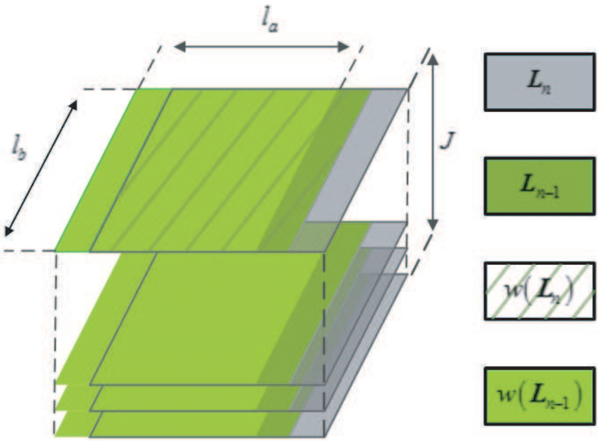


Figure 1: Schematic representation of line-by-line real-time processing.

Authorized licensed use limited to: University of Wyoming Libraries. Downloaded on October 11,2021 at 23:51:05 UTC from IEEE Xplore. Restrictions apply.

4. REAL-TIME KRXD ALGORITHM

The causal version of KRXD (C-KRXD) can be expressed as follows

$$\delta^{KRXD}\left(\Phi\left(\mathbf{L}_n\right)\right) = \left(\mathbf{k}_{\mathbf{L}_n}^T - \mathbf{k}_\mu^T\left(\mathbf{L}\left(n\right)\right)\right)^T \mathbf{K}_w^{-1}\left(\mathbf{L}\left(n\right)\right)\left(\mathbf{k}_{\mathbf{L}_n}^T - \mathbf{k}_\mu^T\left(\mathbf{L}\left(n\right)\right)\right) \tag{7}$$

C-KRXD meets the requirement of real-time causality but misses the timeliness. That is because the computation of $\mathbf{K}_w^{-1}(\mathbf{L}(n))$ can lead to high computational complexity of data processing. In order to reduce the calculated amount, recursively updating the kernel Gram matrix inversion is needed. Using the Gaussian RBF kernel function in Equation (6), we have:

$$\begin{aligned}\mathbf{K}_w\left(\mathbf{L}\left(n-1\right)\right) &= \begin{bmatrix} k\left(\mathbf{L}_{n-l_a-1}, \mathbf{L}_{n-l_a-1}\right) & k\left(\mathbf{L}_{n-l_a-1}, \mathbf{L}_{n-l_a}\right) & \dots & k\left(\mathbf{L}_{n-l_a-1}, \mathbf{L}_{n-2}\right) \\ k\left(\mathbf{L}_{n-l_a}, \mathbf{L}_{n-l_a-1}\right) & k\left(\mathbf{L}_{n-l_a}, \mathbf{L}_{n-l_a}\right) & \dots & k\left(\mathbf{L}_{n-l_a}, \mathbf{L}_{n-2}\right) \\ \vdots & \vdots & \ddots & \vdots \\ k\left(\mathbf{L}_{n-2}, \mathbf{L}_{n-l_a-1}\right) & k\left(\mathbf{L}_{n-2}, \mathbf{L}_{n-l_a}\right) & \dots & k\left(\mathbf{L}_{n-2}, \mathbf{L}_{n-2}\right) \end{bmatrix} \\ &= \begin{bmatrix} \gamma_L & \alpha_L \\ \alpha_L^T & \mathbf{K}_L \end{bmatrix}\end{aligned} \tag{8}$$

where α_L is obtained by $\alpha_L = (k(\mathbf{L}_{n-l_a-1}, \mathbf{L}_{n-l_a}), k(\mathbf{L}_{n-l_a-1}, \mathbf{L}_{n-l_a+1}), \dots, k(\mathbf{L}_{n-l_a-1}, \mathbf{L}_{n-2}))$, and γ_L is given by $\gamma_L = k(\mathbf{L}_{n-l_a-1}, \mathbf{L}_{n-l_a-1})$. Similarly, the $\mathbf{K}_w(\mathbf{L}(n))$ in the sliding causal matrix window at L_n is written as

$$\begin{aligned}\mathbf{K}_w\left(\mathbf{L}\left(n\right)\right) &= \begin{bmatrix} k\left(\mathbf{L}_{n-l_a}, \mathbf{L}_{n-l_a}\right) & k\left(\mathbf{L}_{n-l_a}, \mathbf{L}_{n-l_a+1}\right) & \dots & k\left(\mathbf{L}_{n-l_a}, \mathbf{L}_{n-1}\right) \\ k\left(\mathbf{L}_{n-l_a+1}, \mathbf{L}_{n-l_a}\right) & k\left(\mathbf{L}_{n-l_a+1}, \mathbf{L}_{n-l_a+1}\right) & \dots & k\left(\mathbf{L}_{n-l_a+1}, \mathbf{L}_{n-1}\right) \\ \vdots & \vdots & \ddots & \vdots \\ k\left(\mathbf{L}_{n-1}, \mathbf{L}_{n-l_a}\right) & k\left(\mathbf{L}_{n-1}, \mathbf{L}_{n-l_a+1}\right) & \dots & k\left(\mathbf{L}_{n-1}, \mathbf{L}_{n-1}\right) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{K}_L & \beta_L^T \\ \beta_L & \chi_L \end{bmatrix}\end{aligned} \tag{9}$$

where β_L and χ_L are separately defined by $\beta_L = (k(\mathbf{L}_{n-1}, \mathbf{L}_{n-l_a}), k(\mathbf{L}_{n-1}, \mathbf{L}_{n-l_a+1}), \dots, k(\mathbf{L}_{n-1}, \mathbf{L}_{n-2}))$ and $\chi_L = k(\mathbf{L}_{n-1}, \mathbf{L}_{n-1})$. Taking advantage of the matrix inversion lemma [5], $\mathbf{K}_w^{-1}(\mathbf{L}(n-1))$ and $\mathbf{K}_w^{-1}(\mathbf{L}(n))$ can be expressed as

$$\begin{aligned}\mathbf{K}_w^{-1}\left(\mathbf{L}\left(n-1\right)\right) &= \begin{bmatrix} \gamma_L & \alpha_L \\ \alpha_L^T & \mathbf{K}_L \end{bmatrix}^{-1} = \begin{bmatrix} \gamma_L^{-1} + \gamma_L^{-1}\alpha_L\kappa^{-1}\alpha_L\gamma_L^{-1} & -\gamma_L^{-1}\alpha_L\kappa^{-1} \\ -\kappa^{-1}\alpha_L^T\gamma_L^{-1} & \kappa^{-1} \end{bmatrix} \\ \kappa &= \mathbf{K}_L - \alpha_L^T\gamma_L^{-1}\alpha_L\end{aligned} \tag{10}$$

$$\begin{aligned}\mathbf{K}_w^{-1}\left(\mathbf{L}\left(n\right)\right) &= \begin{bmatrix} \mathbf{K}_L & \beta_L^T \\ \beta_L & \chi_L \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{K}_L^{-1} + \lambda\left(\chi_L + \beta_L^T\lambda\right)^{-1}\lambda^T & \lambda\left(\chi_L + \beta_L^T\lambda\right)^{-1} \\ \left(\chi_L + \beta_L^T\lambda\right)^{-1}\lambda^T & \left(\chi_L + \beta_L^T\lambda\right)^{-1} \end{bmatrix} \\ \lambda &= -\mathbf{K}_L^{-1}\beta_L^T\end{aligned} \tag{11}$$

The $\mathbf{K}_L^{-1}$ in the Equation (10) can be efficiently calculated by using the Woodbury matrix identity [6] as follows:

$$\mathbf{K}_L^{-1} = \left(\kappa + \alpha_L^T\gamma_L^{-1}\alpha_L\right)^{-1} = \kappa^{-1} - \kappa^{-1}\alpha_L^T\left(\gamma_L + \alpha_L\kappa^{-1}\alpha_L^T\right)^{-1}\alpha_L\kappa^{-1} \tag{12}$$

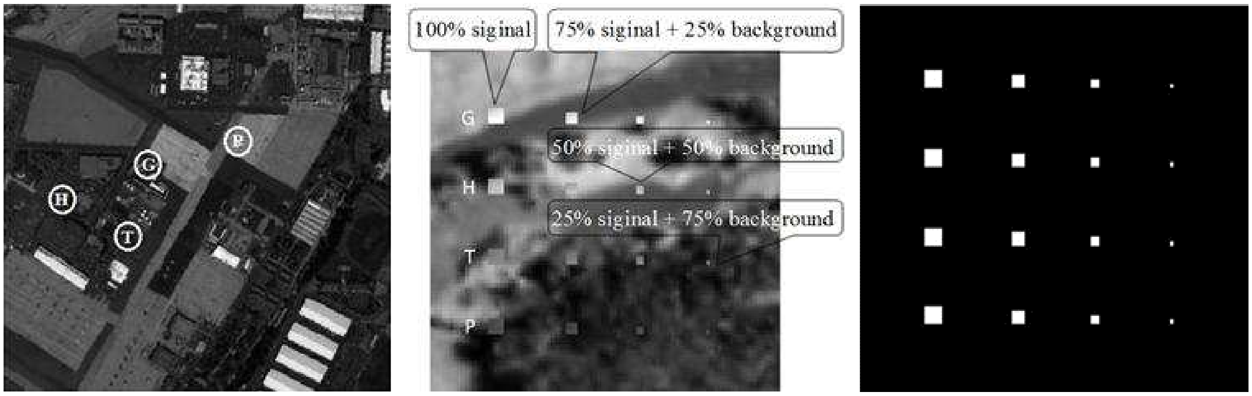
Note that κ^{-1} can be acquired from $\mathbf{K}_w^{-1}(\mathbf{L}(n-1))$, in a word, $\mathbf{K}_w^{-1}(\mathbf{L}(n))$ can be recursively updated by the $\mathbf{K}_w^{-1}(\mathbf{L}(n-1))$, and thus avoiding to calculate $\mathbf{K}_w(\mathbf{L}(n))$ and $\mathbf{K}_w^{-1}(\mathbf{L}(n))$.

5. EXPERIMENTAL RESULTS

In this section, a 126-band synthetic hyperspectral data is used for experiments to investigate the effectiveness and utility of the proposed method. The synthetic data is designed by an airborne hyperspectral image collected by the Airborne Visible Infrared Imaging Spectrometer (AVIRIS) imaging sensor from San Diego airport as shown in Fig. 2(a). The original image has a size of 400 × 400 pixels, with 224 bands, where four targeted spectral signatures (G, H, T, P), marked by circles in it are used to form the synthetic targets. The four targets are used to simulate 16 panels

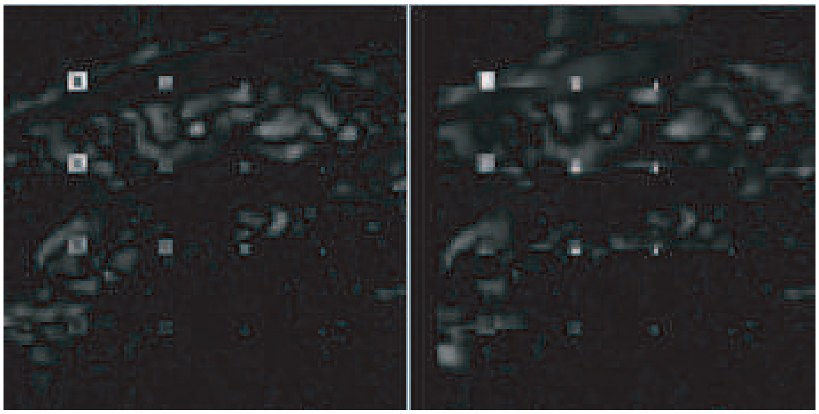
Authorized licensed use limited to: University of Wyoming Libraries. Downloaded on October 11,2021 at 23:51:05 UTC from IEEE Xplore. Restrictions apply.

shown in Fig. 2(b) with 4 panels in each row simulated by same spectral signature and 4 panels in each column with different sizes from left to right (4×4, 3×3, 2×2, 1×1). The background of the synthetic data is a subarea of 90×90 pixels cropped from the HyMap image which obtained by HyMap in the Cook City, MT, USA, with the 126 bands (atmospheric water bands and low SNR bands have been removed) ranging from $0.4\,\mu\text{m}$ to $2.5\,\mu\text{m}$ and a spectral resolution of 3m (discussed in [7]). Fig. 3(c) displays the ground truth map of synthetic image.


P
G
H
T
100% siginal
75% siginal + 25% background
50% siginal + 50% background
25% siginal + 75% background
G
H
T
P

(a) (b) (c)

Figure 2: (a) San Diego Airport image. (b) Synthetic image. (c) Ground truth of synthetic image.



(a) (b)

Figure 3: The grayscale results on the synthetic image. (a) KRXD. (b) RT-KRXD.

By means of cross-validation, the Gaussian RBF kernel parameter c of KRXD and RT-KRXD is set to 5, and the corresponding detection windows are experimentally set to $(11, 5)$ and $(18, 5)$, respectively. As illustrated from grayscale maps in Fig. 3, it is necessary to emphasize that the RT-KRXD provide exactly the same detection results as the KRXD. That is actually true because RT-KRXD is a recursive version of KRXD, there is no information leaked out in the derivation of RT-KRXD. The receiver operating characteristics (ROC) results of two algorithms in Fig. 4 are almost comparable even if KRXD is slightly better than RT-KRXD. Fig. 5 reveals the progressive detection procedures of RT-KRXD, it allows users to see how an anomaly detector performs various degrees of background suppression, and this is also a strong point of real-time processing.

In this paper, the experimental environment was performed on a 64-bit windows 7 operating system with Intel (R) Core (TM) i3-2330M, CPU of 2.20 Hz and 6 GB memory (RAM). All the experiments are conducted five times and averaged to remove computer error. Table 1 shows the total Computing times (CPTs) required by the KRXD and RT-KRXD. With the real-time strategies, it is clearly that RT-KRXD consumes less CPTs than KRXD, and the speedup is gained at least 49-fold in comparison with original detector.

Authorized licensed use limited to: University of Wyoming Libraries. Downloaded on October 11,2021 at 23:51:05 UTC from IEEE Xplore. Restrictions apply.

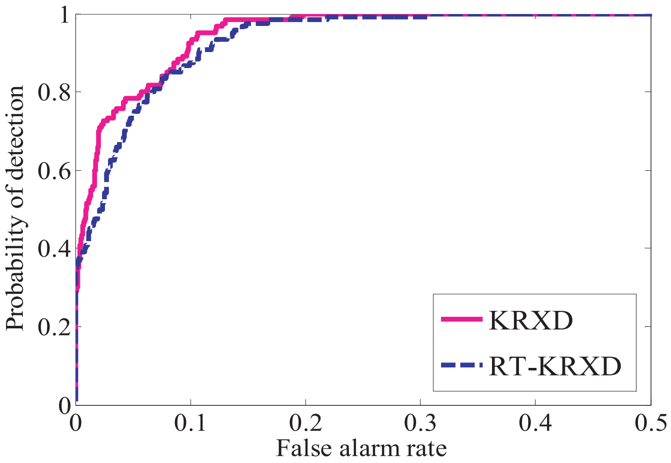


Figure 4: Performance analysis of ROC curves.

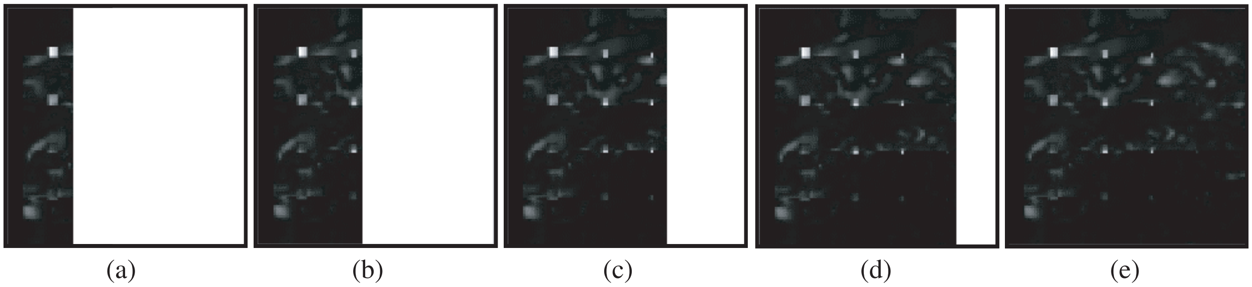


Figure 5: Real-time processing of RT-KRXD. (a) $n = 25$. (b) $n = 40$. (c) $n = 60$. (d) $n = 75$. (e) $n = 90$.

Table 1: Comparison of CPTs for KRXD and RT-KRXD.

	KRXD	RT-KRXD	Speedup
Total CPTs (second)	317.728	6.535	**48.619**

6. CONCLUSION

KRXD which can sufficiently exploit the nonlinear information of datasets caused more and more attention in AD. However, it cannot meet the requirement of real time in actual application due to thekernelization of datasets. This paper develops a real-time version of KRXD, thereby implementing the real-time processing optimization on the basis of original KRXD by solving the problem of causality and timeliness.Moreover, experimental results have demonstrated that the proposed method can remain the detection accuracy unchanged and improve the execution efficiency of the algorithm. Furthermore, how to adaptively choose kernel RBF parameter c and the window size is still one of our future research directions.

ACKNOWLEDGMENT

This work is partially supported by National Natural Science Foundation of China under Grant (Grant No. 61405041 and 61571145), the Key Program of Heilongjiang Natural Science Foundation (Grant No. ZD201216), the Program Excellent Academic Leaders of Harbin (Grant No. RC2013XK-009003) and the Fundamental Research Funds for the Central Universities (Grant No. GK208026016-7). This paper is funded by the International Exchange Program of Harbin Engineering University for Innovation-oriented Talents Cultivation.

REFERENCES

1. Bajorski, P., "Target detection under misspecified models in hyperspectral images," *IEEE Journal of Selected Topics in Applied Earth Observations & Remote Sensing*, Vol. 5, 470–477, 2012.
2. Reed, I. S. and X. Yu, "Adaptive multiple-band CFAR detection of an optical pattern with un-

Authorized licensed use limited to: University of Wyoming Libraries. Downloaded on October 11,2021 at 23:51:05 UTC from IEEE Xplore. Restrictions apply.

known spectral distribution," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. 38, 1760–1770, 1990.

3. Kwon, H., S. Z. Der, and N. M. Nasrabadi, "Adaptive anomaly detection using subspace separation for hyperspectral imagery," *Optical Engineering*, Vol. 42, 3342–3351, 2003.
4. Chen, S. Y., Y. Wang, C. C. Wu, C. Liu, and C. I. Chang, "Real-time causal processing of anomaly detection for hyperspectral imagery," *IEEE Transactions on Aerospace and Electronic System*, Vol. 50, 1511–1534, 2014.
5. Chang, C. I., *Hyperspectral Data Processing: Algorithm Design and Analysis*, John Wiley & Sons, Hoboken, NJ, USA, 2013.
6. Kailath, T., *Linear Systems*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1980.
7. Zhao, C. H., J. W. Li, M. L. Meng, and X. F. Yao, "A weighted spatial-spectral kernel RX algorithm and efficient implementation on GPUs," *Sensors*, Vol. 17, 441–454, 2017.

Authorized licensed use limited to: University of Wyoming Libraries. Downloaded on October 11,2021 at 23:51:05 UTC from IEEE Xplore. Restrictions apply.